3. Under what conditions can we solve (3.2) and be sure to have solved (3.1), i.e. under what conditions do we have $v = w$ and equivalence between the optimal sequential allocation $\{k_{t+1}\}_{t=0}^{\infty}$ and allocations generated by the optimal recursive policy $g(k)$

4. Can we say something about the qualitative features of v and g?

The answers to these questions will be given in the next two sections: the answers to 1. and 2. will come from the Contraction Mapping Theorem, to be discussed in Section 4.3. The answer to the third question makes up what Richard Bellman called the Principle of Optimality and is discussed in Section 5.1. Finally, under more restrictive assumptions we can characterize the solution to the functional equation (v, g) more precisely. This will be done in Section 5.2. In the remaining parts of this section we will look at specific examples where we can solve the functional equation by hand. Then we will talk about competitive equilibria and the way we can construct prices so that Pareto optimal allocations, together with these prices, form a competitive equilibrium. This will be our versions of the first and second welfare theorem for the neoclassical growth model.

3.2.3 An Example

Consider the following example. Let the period utility function be given by $U(c) = \ln(c)$ and the aggregate production function be given by $F(k, n) = k^{\alpha} n^{1-\alpha}$ and assume full depreciation, i.e. $\delta = 1$. Then $f(k) = k^{\alpha}$ and the functional equation becomes

$$v(k) = \max_{0 \leq k' \leq k^{\alpha}} \{\ln(k^{\alpha} - k') + \beta v(k')\}$$

Remember that the solution to this functional equation is an entire function $v(.)$. Now we will apply several methods to solve this functional equation.

Guess and Verify

We will guess a particular functional form of a solution and then verify that the solution has in fact this form (note that this does not rule out that the functional equation has other solutions). This method works well for the example at hand, but not so well for most other examples that we are concerned with. Let us guess

$$v(k) = A + B\ln(k)$$

where A and B are coefficients that are to be determined. The method consists of three steps:

1. Solve the maximization problem on the right hand side, given the guess for v, i.e. solve

$$\max_{0 \leq k' \leq k^{\alpha}} \{\ln(k^{a} - k') + \beta(A + B\ln(k'))\}$$

Obviously the constraints on k' never bind and the objective function is strictly concave and the constraint set is compact, for any given k. The first order condition is sufficient for the unique solution. The FOC yields

$$\begin{aligned} \frac{1}{k^\alpha - k'} &= \frac{\beta B}{k'} \\ k' &= \frac{\beta B k^\alpha}{1+\beta B} \end{aligned}$$

2. Evaluate the right hand side at the optimum $k' = \frac{\beta B k^\alpha}{1+\beta B}$. This yields

$$\begin{aligned} \text{RHS} &= \ln\left(k^a - k'\right) + \beta\left(A + B\ln(k')\right) \\ &= \ln\left(\frac{k^\alpha}{1+\beta B}\right) + \beta A + \beta B \ln\left(\frac{\beta B k^\alpha}{1+\beta B}\right) \\ &= -\ln(1+\beta B) + \alpha\ln(k) + \beta A + \beta B\ln\left(\frac{\beta B}{1+\beta B}\right) + \alpha\beta B\ln(k) \end{aligned}$$

3. In order for our guess to solve the functional equation, the left hand side of the functional equation, which we have guessed to equal LHS$= A + B\ln(k)$ must equal the right hand side, which we just found. If we can find coefficients A, B for which this is true, we have found a solution to the functional equation. Equating LHS and RHS yields

$$\begin{aligned} A + B\ln(k) &= -\ln(1+\beta B) + \alpha\ln(k) + \beta A + \beta B\ln\left(\frac{\beta B}{1+\beta B}\right) + \alpha\beta B\ln(k) \\ (B - \alpha(1+\beta B))\ln(k) &= -A - \ln(1+\beta B) + \beta A + \beta B\ln\left(\frac{\beta B}{1+\beta B}\right) \qquad (3.3) \end{aligned}$$

But this equation has to hold for *every* capital stock k. The right hand side of (3.3) does not depend on k but the left hand side does. Hence the right hand side is a constant, and the only way to make the left hand side a constant is to make $B - \alpha(1+\beta B) = 0$. Solving this for B yields $B = \frac{\alpha}{1-\alpha\beta}$. Since the left hand side of (3.3) is 0, the right hand side better is, too, for $B = \frac{\alpha}{1-\alpha\beta}$. Therefore the constant A has to satisfy

$$\begin{aligned} 0 &= -A - \ln(1+\beta B) + \beta A + \beta B\ln\left(\frac{\beta B}{1+\beta B}\right) \\ &= -A - \ln\left(\frac{1}{1-\alpha\beta}\right) + \beta A + \frac{\alpha\beta}{1-\alpha\beta}\ln(\alpha\beta) \end{aligned}$$

Solving this mess for A yields

$$A = \frac{1}{1-\beta}\left[\frac{\alpha\beta}{1-\alpha\beta}\ln(\alpha\beta) + \ln(1-\alpha\beta)\right]$$

We can also determine the optimal policy function $k' = g(k)$ as

$$\begin{aligned} g(k) &= \frac{\beta B k^\alpha}{1+\beta B} \\ &= \alpha\beta k^\alpha \end{aligned}$$

Hence our guess was correct: the function $v^*(k) = A + B\ln(k)$, with A, B as determined above, solves the functional equation, with associated policy function $g(k) = \alpha\beta k^\alpha$. Note that for this specific example the optimal policy of the social planner is to save a constant fraction $\alpha\beta$ of total output k^α as capital stock for tomorrow and and let the household consume a constant fraction $(1 - \alpha\beta)$ of total output today. The fact that these fractions do not depend on the level of k is very unique to this example and not a property of the model in general. Also note that there may be other solutions to the functional equation; we have just constructed one (actually, for the specific example there are no others, but this needs some proving). Finally, it is straightforward to construct a sequence $\{k_{t+1}\}_{t=0}^\infty$ from our policy function g that will turn out to solve the sequential problem (3.1) (of course for the specific functional forms used in the example): start from $k_0 = \bar{k}_0$, $k_1 = g(k_0) = \alpha\beta k_0^\alpha$, $k_2 = g(k_1) = \alpha\beta k_1^\alpha = (\alpha\beta)^{1+\alpha} k_0^{\alpha^2}$ and in general $k_t = (\alpha\beta)^{\sum_{j=0}^{t-1}\alpha^j} k_0^{\alpha^t}$. Obviously, since $0 < \alpha < 1$ we have that

$$\lim_{t\to\infty} k_t = (\alpha\beta)^{\frac{1}{1-\alpha}}$$

for all initial conditions $k_0 > 0$ (which, not surprisingly, is the unique solution to $g(k) = k$).

Value Function Iteration: Analytical Approach

In the last section we started with a clever guess, parameterized it and used the method of undetermined coefficients (guess and verify) to solve for the solution v^* of the functional equation. For just about any other than the log-utility, Cobb-Douglas production function case this method would not work; even your most ingenious guesses would fail when trying to be verified.

Consider the following iterative procedure for our previous example

1. Guess an arbitrary function $v_0(k)$. For concreteness let's take $v_0(k) = 0$ for all

2. Proceed recursively by solving

$$v_1(k) = \max_{0\le k'\le k^\alpha} \{\ln(k^\alpha - k') + \beta v_0(k')\}$$

   Note that we can solve the maximization problem on the right hand side since we know v_0 (since we have guessed it). In particular, since $v_0(k') = 0$ for all k' we have as optimal solution to this problem

$$k' = g_1(k) = 0 \text{ for all } k$$

   Plugging this back in we get

$$v_1(k) = \ln(k^\alpha - 0) + \beta v_0(0) = \ln k^\alpha = \alpha \ln k$$

3. Now we can solve

$$v_2(k) = \max_{0 \le k' \le k^\alpha} \{\ln(k^\alpha - k') + \beta v_1(k')\}$$

since we know v_1 and so forth.

4. By iterating on the recursion

$$v_{n+1}(k) = \max_{0 \le k' \le k^\alpha} \{\ln(k^\alpha - k') + \beta v_n(k')\}$$

we obtain a sequence of value functions $\{v_n\}_{n=0}^{\infty}$ and policy functions $\{g_n\}_{n=1}^{\infty}$. Hopefully these sequences will converge to the solution v^* and associated policy g^* of the functional equation. In fact, below we will state and prove a very important theorem asserting exactly that (under certain conditions) this iterative procedure converges for any initial guess and converges to the correct solution, namely v^*.

In the first homework I let you carry out the first few iterations in this procedure. Note however, that, in order to find the solution v^* exactly you would have to carry out step 2. above a lot of times (in fact, infinitely many times), which is, of course, infeasible. Therefore one has to implement this procedure numerically on a computer.

Value Function Iteration: Numerical Approach

Even a computer can carry out only a finite number of calculation and can only store finite-dimensional objects. Hence the best we can hope for is a numerical approximation of the true value function. The functional equation above is defined for all $k \ge 0$ (in fact there is an upper bound, but let's ignore this for now). Because computer storage space is finite, we will approximate the value function for a finite number of points only.[4] For the sake of the argument suppose that k and k' can only take values in $\mathcal{K} = \{0.04, 0.08, 0.12, 0.16, 0.2\}$. Note that the value functions v_n then consists of 5 numbers, $(v_n(0.04), v_n(0.08), v_n(0.12), v_n(0.16), v_n(0.2))$

Now let us implement the above algorithm numerically. First we have to pick concrete values for the parameters α and β. Let us pick $\alpha = 0.3$ and $\beta = 0.6$.

1. Make an initial guess $v_0(k) = 0$ for all $k \in \mathcal{K}$

2. Solve

$$v_1(k) = \max_{\substack{0 \le k' \le k^{0.3} \\ k' \in \mathcal{K}}} \left\{\ln\left(k^{0.3} - k'\right) + 0.6 * 0\right\}$$

[4] In this course I will only discuss so-called finite state-space methods, i.e. methods in which the state variable (and the control variable) can take only a finite number of values. Ken Judd, one of the world leaders in numerical methods in economics teaches an exellent second year class in computational methods, in which much more sophisticated methods for solving similar problems are discussed. I strongly encourage you to take this course at some point of your career here in Stanford.

This obviously yields as optimal policy $k'(k) = g_1(k) = 0.04$ for all $k \in \mathcal{K}$ (note that since $k' \in \mathcal{K}$ is required, $k' = 0$ is not allowed). Plugging this back in yields

$$\begin{aligned}
v_1(0.04) &= \ln(0.04^{0.3} - 0.04) = -1.077 \\
v_1(0.08) &= \ln(0.08^{0.3} - 0.04) = -0.847 \\
v_1(0.12) &= \ln(0.12^{0.3} - 0.04) = -0.715 \\
v_1(0.16) &= \ln(0.16^{0.3} - 0.04) = -0.622 \\
v_1(0.2) &= \ln(0.2^{0.3} - 0.04) = -0.55
\end{aligned}$$

3. Let's do one more step by hand

$$v_2(k) = \left\{ \max_{\substack{0 \leq k' \leq k^{0.3} \\ k' \in \mathcal{K}}} \ln\left(k^{0.3} - k'\right) + 0.6 v_1(k') \right\}$$

Start with $k = 0.04$:

$$v_2(0.04) = \max_{\substack{0 \leq k' \leq 0.04^{0.3} \\ k' \in \mathcal{K}}} \left\{ \ln\left(0.04^{0.3} - k'\right) + 0.6 v_1(k') \right\}$$

Since $0.04^{0.3} = 0.381$ all $k' \in \mathcal{K}$ are possible. If the planner chooses $k' = 0.04$, then

$$v_2(0.04) = \ln\left(0.04^{0.3} - 0.04\right) + 0.6 * (-1.077) = -1.723$$

If he chooses $k' = 0.08$, then

$$v_2(0.04) = \ln\left(0.04^{0.3} - 0.08\right) + 0.6 * (-0.847) = -1.710$$

If he chooses $k' = 0.12$, then

$$v_2(0.04) = \ln\left(0.04^{0.3} - 0.12\right) + 0.6 * (-0.715) = -1.773$$

If $k' = 0.16$, then

$$v_2(0.04) = \ln\left(0.04^{0.3} - 0.16\right) + 0.6 * (-0.622) = -1.884$$

Finally, if $k' = 0.2$, then

$$v_2(0.04) = \ln\left(0.04^{0.3} - 0.2\right) + 0.6 * (-0.55) = -2.041$$

Hence for $k = 0.04$ the optimal choice is $k'(0.04) = g_2(0.04) = 0.08$ and $v_2(0.04) = -1.710$. This we have to do for all $k \in \mathcal{K}$. One can already see that this is quite tedious by hand, but also that a computer can do this quite rapidly. Table 1 below shows the value of

$$\left(k^{0.3} - k'\right) + 0.6 v_1(k')$$

for different values of k and k'. A $*$ in the column for k' that this k' is the optimal choice for capital tomorrow, for the particular capital stock k today

Table 1

$k \backslash k'$	0.04	0.08	0.12	0.16	0.2
0.04	-1.7227	-1.7097^*	-1.7731	-1.8838	-2.0407
0.08	-1.4929	-1.4530^*	-1.4822	-1.5482	-1.6439
0.12	-1.3606	-1.3081^*	-1.3219	-1.3689	-1.4405
0.16	-1.2676	-1.2072^*	-1.2117	-1.2474	-1.3052
0.2	-1.1959	-1.1298	-1.1279^*	-1.1560	-1.2045

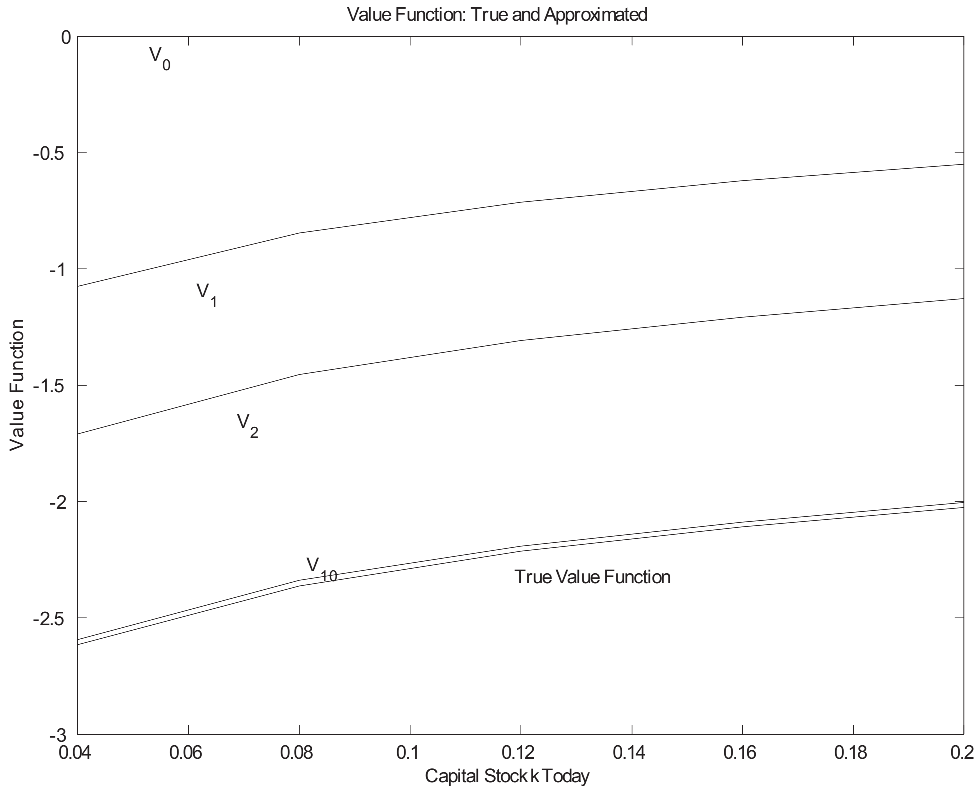
Hence the value function v_2 and policy function g_2 are given by

Table 2

k	$v_2(k)$	$g_2(k)$
0.04	-1.7097	0.08
0.08	-1.4530	0.08
0.12	-1.3081	0.08
0.16	-1.2072	0.08
0.2	-1.1279	0.12

In Figure 3.2.3 we plot the true value function v^* (remember that for this example we know to find v^* analytically) and selected iterations from the numerical value function iteration procedure. In Figure 3.2.3 we have the corresponding policy functions.

We see from Figure 3.2.3 that the numerical approximations of the value function converge rapidly to the true value function. After 20 iterations the approximation and the truth are nearly indistinguishable with the naked eye. Looking at the policy functions we see from Figure 2 that the approximating policy function do not converge to the truth (more iterations don't help). This is due to the fact that the analytically correct value function was found by allowing $k' = g(k)$ to take any value in the real line, whereas for the approximations we restricted $k' = g_n(k)$ to lie in $\mathcal{K}$. The function g_{10} approximates the true policy function as good as possible, subject to this restriction. Therefore the approximating value function will not converge exactly to the truth, either. The fact that the value function approximations come much closer is due to the fact that the utility and production function induce "curvature" into the value function, something that we may make more precise later. Also note that we we plot the true value and policy function only on $\mathcal{K}$, with MATLAB interpolating between the points in $\mathcal{K}$, so that the true value and policy functions in the plots look piecewise linear.

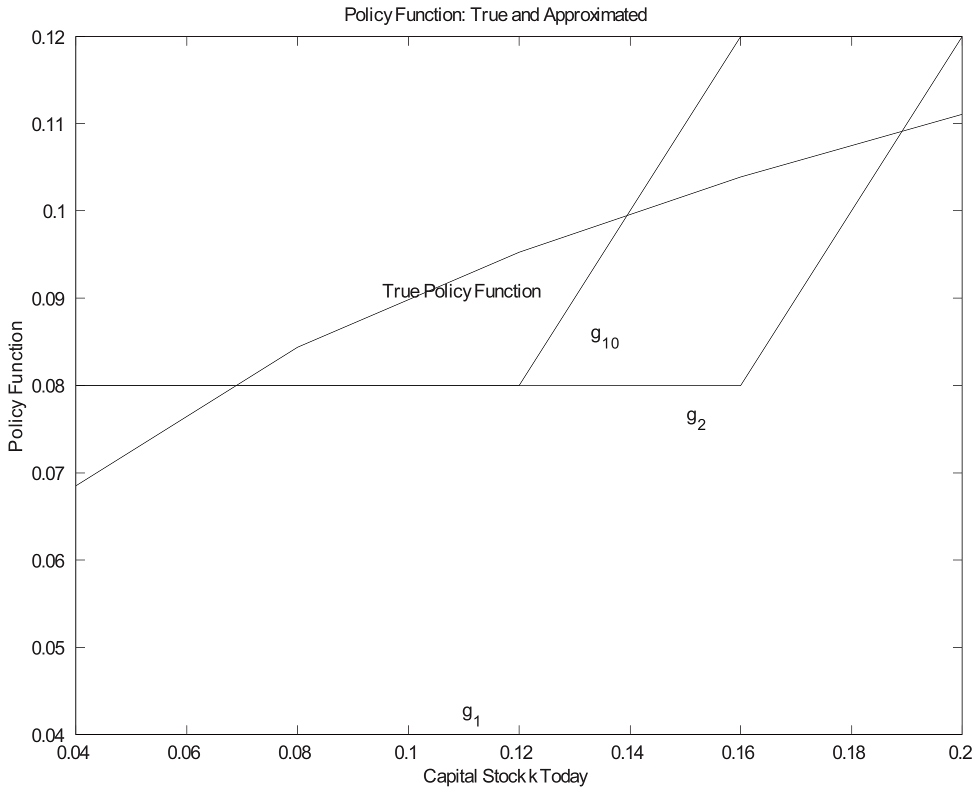


3.2.4 The Euler Equation Approach and Transversality Conditions

We now relate our example to the traditional approach of solving optimization problems. Note that this approach also, as the guess and verify method, will only work in very simple examples, but not in general, whereas the numerical approach works for a wide range of parameterizations of the neoclassical growth model. First let us look at a finite horizon social planners problem and then at the related infinite-dimensional problem

The Finite Horizon Case

Let us consider the social planner problem for a situation in which the representative consumer lives for $T < \infty$ periods, after which she dies for sure and



the economy is over. The social planner problem for this case is given by

$$
\begin{aligned}
w^T(\bar{k}_0) &= \max_{\{k_{t+1}\}_{t=0}^T} \sum_{t=0}^{T} \beta^t U(f(k_t) - k_{t+1}) \\
0 &\leq k_{t+1} \leq f(k_t) \\
k_0 &= \bar{k}_0 > 0 \text{ given}
\end{aligned}
$$

Obviously, since the world goes under after period T, $k_{T+1} = 0$. Also, given our Inada assumptions on the utility function the constraints on k_{t+1} will never be binding and we will disregard them henceforth. The first thing we note is that, since we have a finite-dimensional maximization problem and since the set constraining the choices of $\{k_{t+1}\}_{t=0}^T$ is closed and bounded, by the Bolzano-Weierstrass theorem a solution to the maximization problem exists, so that $w^T(\bar{k}_0)$ is well-defined. Furthermore, since the constraint set is convex and we assumed that U is strictly concave (and the finite sum of strictly concave functions is strictly concave), the solution to the maximization problem is unique and the first order conditions are not only necessary, but also sufficient.